WHAT IS CLAIMED IS:

1.   A computer program product for performing format conversions, tangibly
stored on a computer-readable medium, comprising instructions operable to cause a
programmable processor to:

       identify a first representation of a persistent object and first format indicator
associated with the persistent object;

       identify a current format indicator;

       apply a conversion engine, wherein the conversion engine takes as inputs a first
schema referenced by the first format indicator, a second schema referenced by the
current format indicator and the first representation of the persistent object and wherein
the conversion engine generates a second representation in a format referenced by the
current format indicator; and

       read the second representation to recreate the persistent object.

2.   The computer program product of claim 1, wherein the format indicators
are format numbers.

3.   The computer program product of claim 1, wherein the schema are
contained in a secondary program controlled by a primary program.

4.   The computer program product of claim 3, wherein the secondary program
is a plug-in and the primary program is an application program associated with the plug-
in.

5.   The computer program product of claim 1, wherein the conversion engine
cooperates with a second conversion engine capable of converting formats without
reference to the schema.

6.   The computer program product of claim 5, wherein the conversion engines
are part of a single conversion service contained in a single application program.

7. A file conversion system, comprising:

a primary program;

a secondary program called by the primary program;

at least one file associated with the primary program, the at least one file including a first representation of a persistent object created by the secondary program;

a conversion service contained in the primary program;

a set of schemas contained in the secondary program, the schemas describing the format in which previous versions of the secondary program wrote data into the at least one file and the format in which the current version of the secondary program writes data;

wherein the conversion service is capable of converting the first representation of a persistent object from a first format to a second format based on the format information provided by the set of schemas.

8. The system of claim 7, wherein the secondary program is a plug-in.

9. The system of claim 8, wherein the conversion service cooperates with a second conversion service capable of converting data from one format to another without reference to the schema.

10. The system of claim 9, wherein the conversion services are part of a single hybrid conversion service contained in a single application program.

11. A method for performing format conversions, comprising:

finding a first representation of a persistent object and first format indicator associated with the persistent object;

finding a current format indicator;

applying a conversion engine, wherein the conversion engine takes as inputs a first schema referenced by the first format indicator, a second schema referenced by the current format indicator and the first representation of the persistent object and wherein

17

the conversion engine generates a second representation in a format referenced by the current format indicator; and

reading the second representation to recreate the persistent object.

5   12.   The method of claim 11, wherein the format indicators are format numbers.

13.   The method of claim 11, wherein the schema are contained in a secondary program controlled by a primary program.

10

14.   The method of claim 13, wherein the secondary program is a plug-in and the primary program is an application program associated with the plug-in.

15.   The method of claim 11, wherein the conversion engine cooperates with a
15   second conversion engine capable of converting formats without reference to the schema.

16.   The method of claim 15, wherein the conversion engines are part of a single conversion service contained in a single application program.

20   17.   A method for performing format conversions, comprising:

a)   read a file comprising a plurality of persistent objects written by one or more secondary programs;

b)   identify a current format indicator associated with a format compatible with a current version of a secondary program;

25   c)   identify a first format indicator associated with a first object created by a version of the secondary program;

d)   compare the format indicators to determine whether the persistent object is in a format compatible within the current version of the secondary program;

e)   where the format of the persistent object is not compatible, apply a
30   conversion engine in a primary program to convert the data of the persistent object to a

18

compatible format based upon format information that is associated with the format indicators and contained within the secondary program;

      f)   identify another format indicator associated with another object; and

      g)   repeat steps (d)-(f) to convert the remaining persistent objects to formats

5   compatible with the current versions of the secondary programs.

      18.   A method for performing reverse format conversions, comprising:

      finding a current format indicator;

      querying a conversion service associated with a secondary program to identify a

10   target format indicator associated with a format in which a previous version of the secondary program writes data;

      applying a conversion engine in a primary program associated with the secondary program, wherein the conversion engine takes as inputs i) current and target schema referenced by the format indicators and stored within the secondary program and ii) a

15   first representation of a persistent object,

      whereby the conversion engine generates a second representation of the persistent object in a format referenced by the target format indicator.

20

19